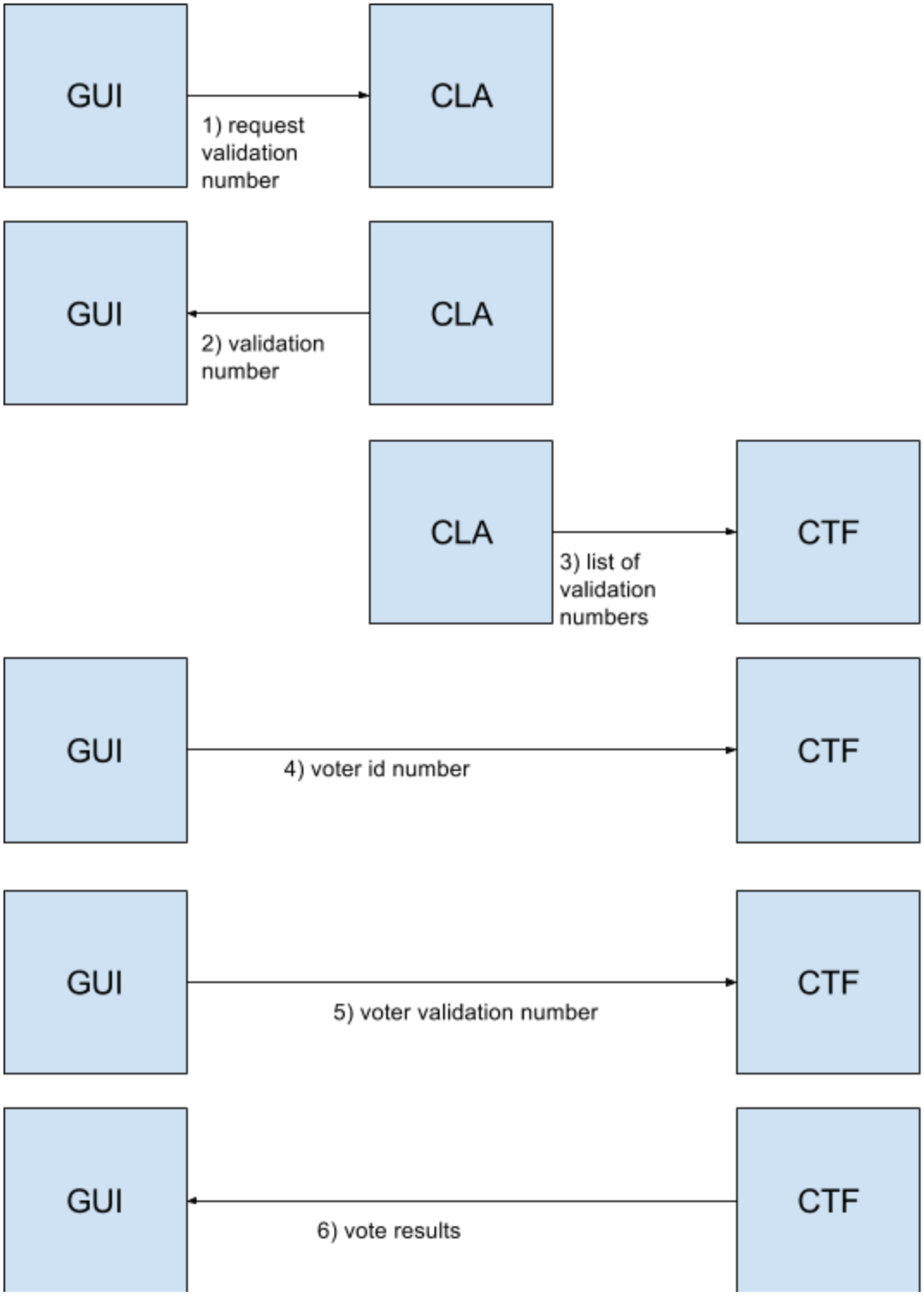
Nathanial Schaffner
Virtual Election Booth Final Project Report

Design Notes - Protocol Steps

1. A voter use the CLI frontend to interface with Central Legitimization Agency (CLA) to register her/his name and receive a voter validation number (VVN).
2. The CLA checks to make sure that the voter has not already registered for a VVN.
    a. If s/he has not, the CLA generates a random number, makes sure that it doesn't collide with an already-assigned value, adds it to the list of VVN's, and returns it to the CLI frontend.
    b. If the voter's name has already been used to request a VVN, the CLI frontend will receive an error message instead.
3. The CLI frontend is used to contact the CLA, and requests transfer of VVN's to the Central Tabulating Facility (CTF). If both the CLA and the CTF servers are online, the VVN's are transferred one at a time (with confirmation of insertion)
4. The CLI frontend allows users to input a random voter identification number, and sends it to the CTF along with the voter's VVN and voter's choice
5. The CTF verifies the information it is provided
    a. If the identification number has already been used, the CTF will return a response to the CLI frontend indicating that a different number needs to be used
    b. If the VVN has already been used to vote, the CTF will send an error message indicating this to the CLI frontend
    c. If the VVN is not in the list transferred from the CLA, the CTF will send an error message to the CLI frontend
    d. If the voter id number has not been used, and the VVN is in the list transferred to the CTF from the CLA, and the VVN has not been used to vote, the CTF will store the vote
6. After all votes have been received, the CTF makes the results available to the CLI frontend (with the 'view' command). For each option (A or B) the results information shows the random identification numbers that voted for each choice, as well as a list of all VVN's that participated in the vote

Design Notes - Protocol Diagram



Design Notes - System Design

- Central Legitimization Agency
    - Implemented in Java
    - Command line server
    - Accessed by frontend over SSL-encrypted connection
- Central Tabulating Facility
    - Implemented in Java
    - Command line server
    - Accessed by CLA and frontend over SSL-encrypted connection
- CLI frontend
    - Implemented in Java
    - Command line menu based system
    - Accesses CLA and CTF over SSL-encrypted connection

Design Notes - Security Requirements

- Only authorized voters can vote
    - Voters use the CLI frontend to communicate with the CLA to register the voters name and generate a VVN
- Nobody can vote more than once
    - The CLA manages voter validation numbers, which can only be used once
    - The CTF manages use of VVN's and voter identification numbers, both of which can only be used to cast a vote once
- Nobody can determine for whom anyone else voted
    - All voter id numbers are publically available after the vote is over, but only the individual users know their identification numbers
    - All participating VVN's are listed after the vote, but VVN's 1) should be confidential to end users and 2) are not associated with the voter's choice
- Nobody's vote can be duplicated
    - Each voter has a unique VVN, and a unique voter identification number. The CLA and the CTF prevent duplication of votes by managing these numbers
- Every voter can see that his vote was counted in the final tally
    - After the vote is completed, the user can issue a "view" command to see what voter identification numbers chose what candidate, and what VVN's participated in the vote overall
    - Each voter should be able to see his VVN and his voter identification number
- Everyone knows who voted and who didn't
    - The "view" command shows all the participating VVN's, so that everyone can see what voters participated and what numbers are absent, while still protecting the confidentiality and anonymity of the voters

Implementation Notes

- All network communication is by way of secure sockets using the javax.net.ssl libraries, using the keytool command line utility to create a keystore/truststore shared between the frontend and backends
- Originally, the plan was to use Swing to implement the GUI frontend, but after getting into the work of the project, I opted for a more simple command line interface

Performance Measurements

- The number of voters is set at 1000, based on the way that the numbers are generated. The source code can be modified for a higher or lower limit, based on multiplying a random value between 0 and 1 by a factor of 10, and converting from double to integer.
- The choices for voting are ‘A’ or ‘B’, but the source code can be modified to allow any number of choices
- The only data structured used are arrayLists, so space and time requirements are very efficient.